

2017 年度 修士論文

入力値再利用型操作手順提示システムの自動生
成に関する研究

2018 年 1 月 30 日（火）提出

指導：深澤 良彰 教授

早稲田大学 基幹理工学研究科 情報理工・情報通信専攻

学籍番号：5116F065-5

中村 史門

目次

第 1 章	はじめに	1
第 2 章	関連研究	3
2.1	LemonAid	3
2.2	Skillometers.....	3
2.3	Ehelp.....	4
第 3 章	既存の操作支援ツール	5
3.1	オンラインマニュアル	5
3.2	オンラインヘルプ	6
3.3	状況依存ヘルプ	7
3.4	チュートリアル	8
3.5	アニメデモンストレーション	9
3.6	入力予測変換	10
第 4 章	本研究の特徴	12
4.1	入力再利用	12
4.2	ユーザの状況に合わせた操作方法の提示	13
4.3	開発者のコスト削減	13
4.4	学習効果	14
第 5 章	操作支援システムの動作	15
5.1	操作支援システムの利用	15
5.1.1	ユーザによる入力再利用	17
5.1.2	ユーザによる操作手順の参照	18
5.2	入力再利用の仕組み	18
5.2.1	入力データの保持	19
5.2.2	入力データの再利用	19
5.3	操作手順提示の仕組み	19
5.3.1	トップ画面からゴールまでの経路導出	20
5.3.2	ユーザの現在画面の把握と履歴保持	24
5.3.3	ユーザの現在画面からゴールまでの手順の導出	24
第 6 章	本研究の詳細	27
6.1	オントロジの作成	27
6.2	入力項目とラベル名の対応付け	28

6.3	アクティビティ図の記述	29
6.3.1	アクティビティ図構成要素	29
6.3.2	アクティビティ図への追加記述	30
6.4	アクティビティ図の解析	31
6.5	開発者による機能名と提示文の記述	32
6.6	操作支援システムのソースコード生成	33
6.6.1	操作支援システムのインタフェース	33
6.6.2	入力再利用を行う機能	33
6.6.3	現在画面把握機能	34
6.6.4	経路導出機能	34
第7章	評価	35
7.1	被験者	35
7.2	手順と装置	35
7.3	結果	36
第8章	考察	40
第9章	おわりに	42
謝辞	43
参考文献	44
研究業績	47

第1章 はじめに

情報社会化が進んでいる近年、多種多様なアプリケーションや Web サービスが開発され、世の中の生活と密接な関係になっている。こうした発展のもと、複雑化し大規模なシステムも存在している。しかし、こうしたシステムはユーザにとって使いにくいと感じられたり、機能を利用する上で多くの操作を必要とすることがあるため、ユーザのミスも増え、ミスをする手戻りが必要になることがある。そうすると、一度入力した内容を再入力しなければならないことも多く、ユーザにとっての操作負担が大きくなる。ミスをした時の手戻りの際に、正しい手順に進むためにはオンラインヘルプやマニュアルなどを参照するが、これらはトップ画面からの操作方法を提示していることが多い[1][2][3]。そうすると、仮に、正しい手順に進んだときに、操作ミスをした過程で入力したものと同様の入力項目があったとしても、再入力が必要になる。しかし、既存のユーザビリティに関する研究では、情報を再利用することや、操作数をより少なくすることでユーザの手間を削減することが提案されている[4][5][6]、このように、同じ内容の入力を繰り返させることは、ユーザの操作負担の観点で問題がある。

そこで本研究では、操作ミスで正しい手順とは異なる手順に進んだとしても、入力した内容をできるだけ無駄にせず、正しい手順に進むことができるような入力値再利用型操作手順提示システムを提案する。具体的には、ユーザが操作の過程で入力してきた入力項目のラベル名と類似した入力項目に対して、一度入力した内容を再利用することで、ユーザの操作負担を削減する。また、ユーザの現在の操作ステップから、目的の機能の操作ステップまで戻り、完了するまでの手順を提示することで、再入力による操作負担を削減する。本論文では、前者を入力再利用機能、後者を操作手順提示機能と呼ぶことにする。本システムの生成には、アプリケーションの各入力項目名を概念として含んだオントロジと、自動生成に必要な情報が追加記述されたアクティビティ図によって行われる。追加記述される情報としては、各アクティビティに該当する画面オブジェクト名と、対象アプリケーションの各機能名、また、各機能を完了するために各画面で行うべき操作内容がある。

本稿は 8 章で構成されている。2 章では本研究の関連研究について触れ、3 章では既存の操作支援ツールについて述べる。4 章では本研究の特徴、5 章では本研究が提案する操作支援システムの動作、6 章では操作支援システムを生成する

ための手順について述べ、7 章では評価を行い、8 章で考察を行う。最後に 9 章で本論文の総括と、本研究の今後の展望について述べる。

第2章 関連研究

ユーザの操作支援に対しては、様々な形の操作支援システムが提案されている。本研究では、入力再利用機能と操作手順提示機能を持った操作支援システムにより、ユーザの操作負担の削減を目指している。本研究とのこれらの関連研究について述べる。

2.1 LemonAid

Parmit らは、LemonAid というヘルプシステムを提案している[7][8]。これは、既存のアプリケーションに統合されるもので、ユーザがアプリケーションのインターフェース要素を選択することで、それに関連した質問と回答を得るというものである。LemonAid は、クラウドソーシングヘルプシステムであり[9]、すべての質問と回答はアプリケーションを使用しているユーザによるものである。インターフェース要素に着目したヘルプは有用であるが、このヘルプシステムから得られるヘルプ情報が不特定多数のユーザ次第であるので、利用するユーザに限られるようなアプリケーションでは、ユーザの絶対数が少ないため、有用となるヘルプ情報も少なくなるといった可能性が大きく、LemonAid の有効性が現れるアプリケーションは限られる。本システムは、アプリケーションで利用出来る機能を全て網羅するため、ヘルプ情報が少なくなるということがない。

2.2 Skillometers

Sylvain らは、ユーザがアプリケーションの機能を利用する際、ショートカットキーの有用性や使用方法などを可視化することで、ユーザに効率的な操作方法を提示する Skillometers を提案している[10]。例えば、環境が MacOS、Web ブラウザが FireFox の場合、新しくウィンドウを新規に開くコマンドは、カーソルを利用してメニューバーの「ファイル」から「新規ウィンドウ」ボタンを押下するか、あるいは、ショートカットキーを利用すれば「Command + N」である。Skillometers は、両者の操作時間を比較し、より早く実行できるショー

トカットキーによる操作がカーソルでの操作に比べてどれほど有効であるか、また、そのショートカットキー操作方法を提示する。このように、ショートカットキーによるコマンドが用意されてある機能については非常に有用であるが、そうではない機能や、あるいは多くの操作が必要な機能については考慮していない。多くの操作や入力を要する大規模で複雑なアプリケーションにおいては、本研究で提案する操作支援システムによる支援の方が有用であると考えられる。

2.3 Ehlp

ユーザの間違いや非効率的な操作を感知して能動的に支援を行うシステムについての研究も行われている[11][12]。能動的な操作支援システムでは、ユーザが望まない支援を受ける可能性があるため、受け入れられないことも多い[1]。システム側ではなく、ユーザ側に制御があるものがユーザには好まれる[1]。本研究で提案する操作支援システムは、ユーザ自身の操作によって支援が始まるため、ユーザが望まない支援とはならない。

第3章 既存の操作支援ツール

今日では、操作支援ツールとして様々なものがある。ユーザは、アプリケーションの操作で手詰まりになってしまった場合や、操作の簡易化を図るための操作支援ツールを活用し、問題解決を図る。主に用いられる既存の操作支援ツールは、以下のものが挙げられる。

- (1) オンラインマニュアル
- (2) オンラインヘルプ
- (3) 状況依存ヘルプ
- (4) チュートリアル
- (5) アニメデモンストレーション
- (6) 入力予測変換

がある[1][13][14]。以下にこれらについて説明する。

3.1 オンラインマニュアル

家電や様々な製品には、冊子になっているマニュアルがある場合が多い。そのマニュアルには、その商品の設定方法や使用方法、機能に関する様々なことが記述されている。ソフトウェアにおいては、これをオンラインマニュアルという形態で取ることがある。オンラインマニュアルは、ソフトウェアとは独立したインターフェースであり、PDF ファイルのようなドキュメント形式のものである場合が多い。ユーザはこのオンラインマニュアルをウェブ上で利用することが可能である。また、電子的なデータであるので用語の検索をかけることが可能である。基本的にはソフトウェアを利用する前に参照するものである。図 1 はユーザマニュアルの例である。このように、操作や機能に関するマニュアルが記載されている。

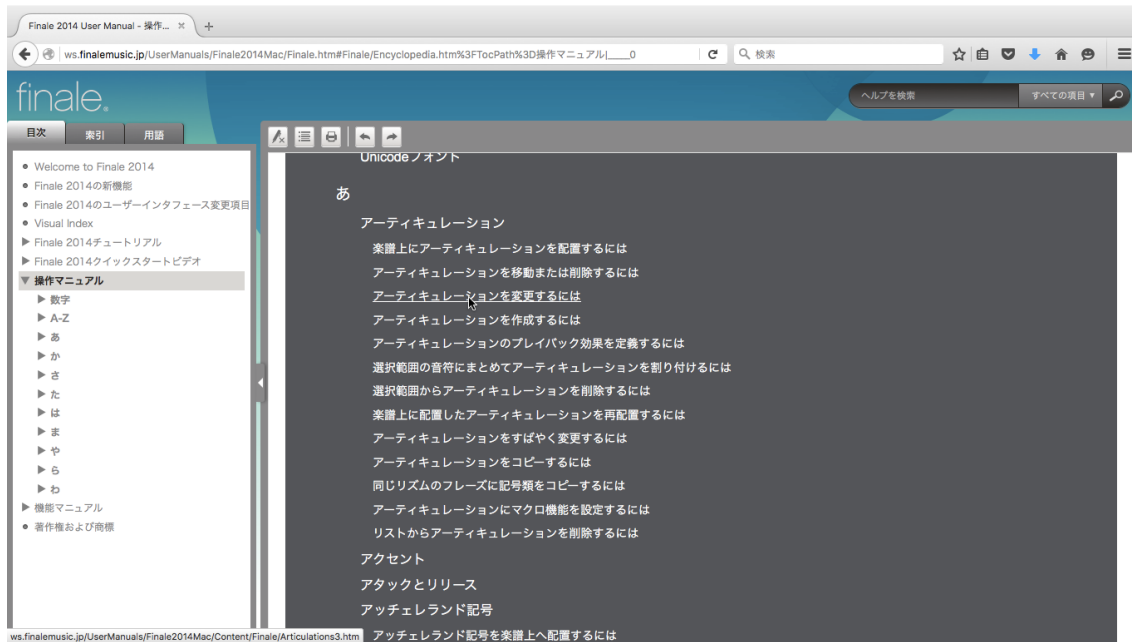


図 1 Finale 2014 のユーザマニュアル
(<http://ws.finalemusic.jp/UserManuals/Finale2014Win/Finale.htm>)

3.2 オンラインヘルプ

現代のモバイル端末によく見られる形態である。一般的なタイプでは、ソフトウェアのインタフェース内にヘルプアイテムが設置されており、ユーザがそこにアクセスすると、ソフトウェアのインタフェース要素やアクションに関する簡易な説明を別のウィンドウで閲覧することができ、ユーザが問題に直面したときに対処できるように、特定のトピックについての記述がなされている。また、FAQ などともここから閲覧できることが多い。図 2 はオンラインヘルプの例である。Word のツールバーにヘルプボタンが設置されており、そこからユーザはヘルプ情報を参照することが可能である。

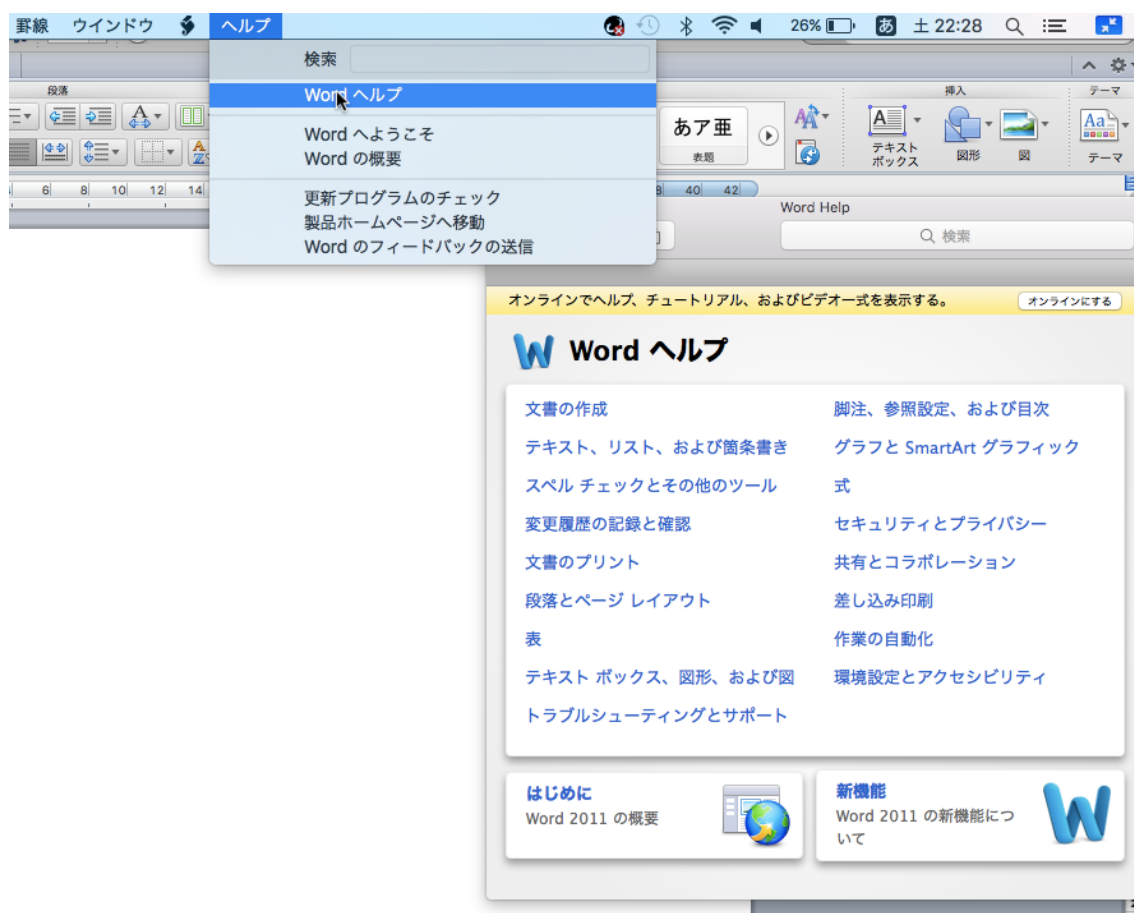


図 2 Word for mac 2011 のオンラインヘルプ

3.3 状況依存ヘルプ

この形態の操作支援ツールは、ユーザの現在の状態を利用してヘルプ情報をユーザに提示するものであり、非常に強力なツールである。しばしばパソコンのソフトウェアにおいて活用されているツールチップというものは状況依存ヘルプである。カーソル情報をシステムが参照し、カーソルが照準を合わせているインタフェース要素の説明を示すポップアップテキストを、ユーザに提示するというものである。Microsoft の Word での状況依存ヘルプの例を図 3 に示す。カーソル下にあるポップアップテキスト「この文書を保存します」が状況依存ヘルプである。状況依存ヘルプは、アプリケーションインターフェース上のそれぞれのオブジェクトが何を意味しているかをユーザに示すのに非常に良い手段である。

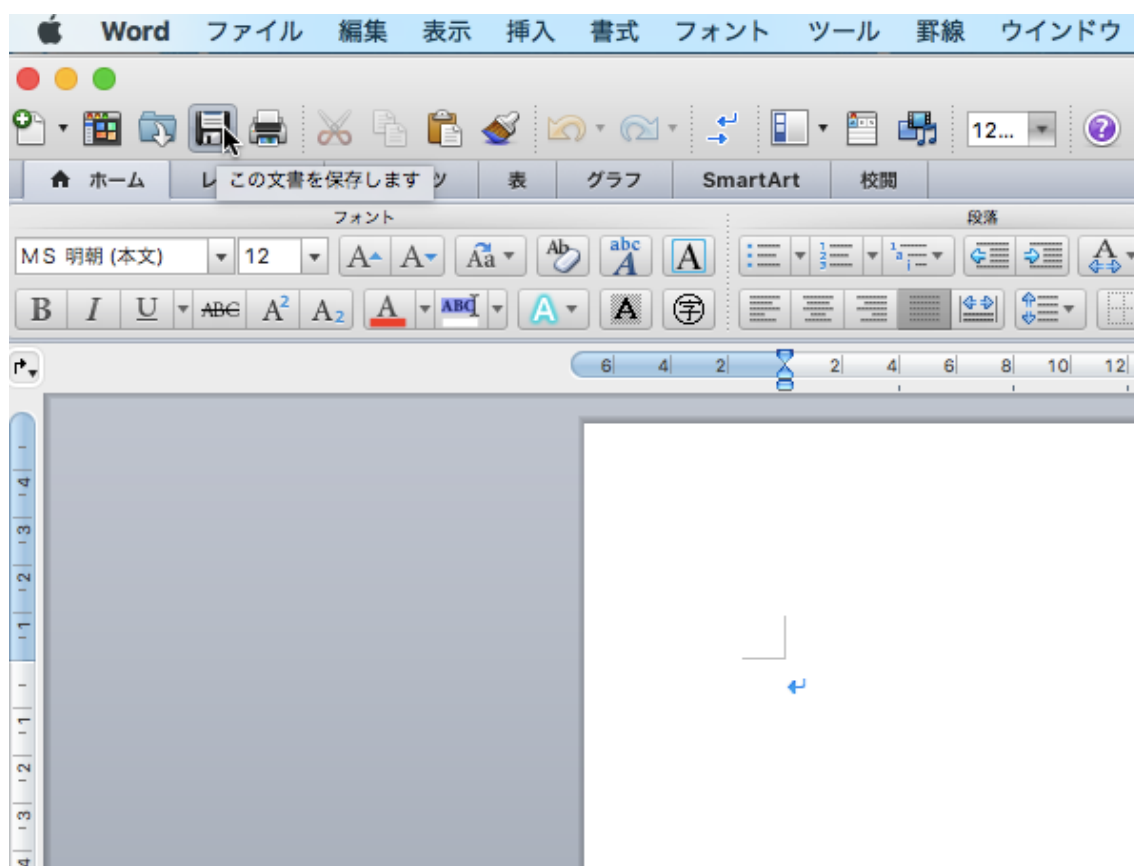


図 3 Word for mac 2011 での状況依存ヘルプ

3.4 チュートリアル

チュートリアルは実践型の操作支援ツールである。開発者はアプリケーションの基本的なタスクをチュートリアルで設定する。そして、アプリケーションを用いてそのタスクをどのように達成していくかをステップごとにユーザーに提示しながら、実際にユーザーが操作を行って操作方法を学習するものである。特に、アプリケーションゲームではしばしばチュートリアルは用いられており、そのゲーム特有の操作や新しい概念をユーザーに習得してもらう際に非常に有用である。図 4 は Word 内で「新しい空白の文書を作成する」をタスクとして設定したチュートリアルの例である。

1. 新しい空白の文書を作成する

Word では、文書を作成して保存できます。空白の文書、既に保存されている文書、またはテンプレートを使用できます。Word を起動すると Word 文書ギャラリーが開き、さまざまなテンプレートや最近使用した文書にすばやくアクセスできます。

Word 文書ギャラリーを使用して、空白の文書を開いてみましょう。

- 1 Word をまだ起動していない場合は、Dock で [Microsoft Word] をクリックします。



- 2 Word 文書ギャラリーの左側の [テンプレート] で、[すべて] をクリックします。

💡 ヒント 文書ギャラリーが表示されない場合は、Word を終了して再起動します。



- 3 [Word 文書] をクリックし、[選択] をクリックします。



図 4 Word for mac 2011 のチュートリアル

3.5 アニメデモンストレーション

アニメデモンストレーションでは、スライドショーや、スクリーンキャプチャアニメーション、デバイスを使用しているビデオなどを用いてユーザに使用方法の説明を行う。アニメーションやグラフィック、サウンドなどを使いながらシステムの機能を見せるものである。ステップごとに説明を行い、自動的に進めるか自分で進めていくかを選択できるようにすれば受動的、能動的なユーザの満足度を得られる。図 5 は Word の新機能に関するアニメデモンストレーションであり、動画で説明を行なっている。

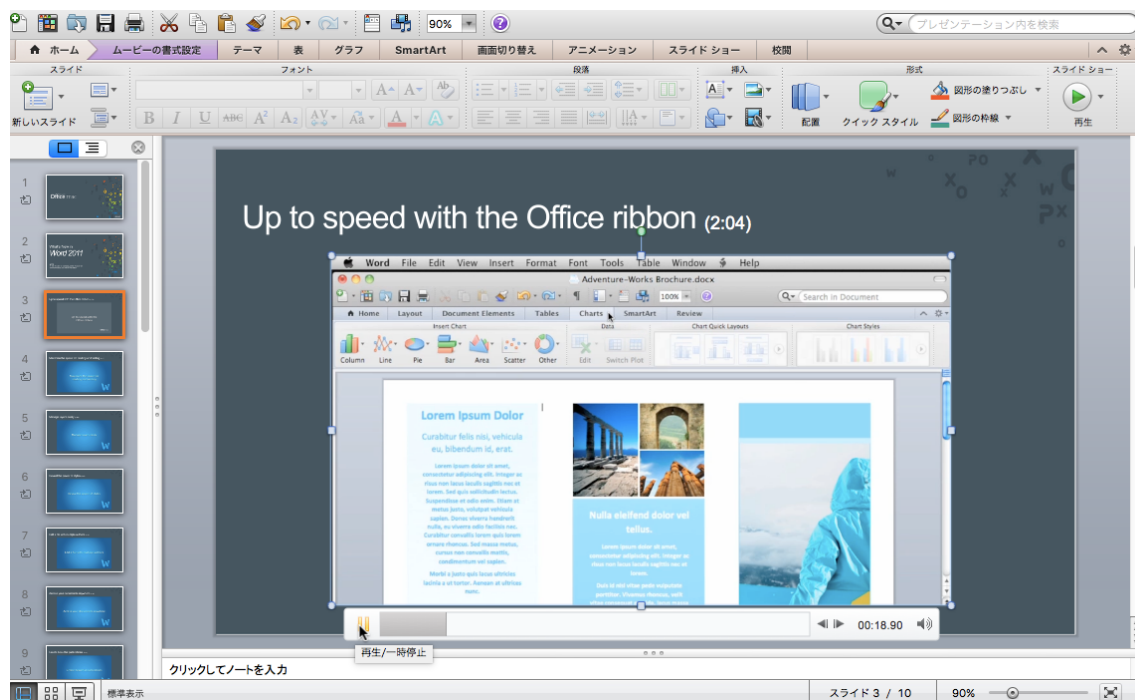


図 5 Word for Mac 2011 の新機能に関するビデオ

3.6 入力予測変換

端末におけるテキスト入力の際に、ユーザが入力した文字列から、テキストを予測変換する機能である。特に、モバイル端末では、入力操作をより容易にするために、入力予測変換は非常に有用である。図 6 は iOS のメールアプリケーションにて、「こんに」と入力した時の状態である。左下側に入力候補が表示され、残りの文字を入力しなくても、一回のタップ操作で入力可能である。

キャンセル	新規メッセージ	送信
-------	---------	----

件名:

こんに

→	こんに	今日	<	あ	か	さ	⌫
↶	こんにちは		>	た	な	は	次候補
ABC	こんにして	コンニ	😊	ま	や	ら	確定
🌐 🎤	こんな	コンに	↷	っ°	わ_	、。?!	

図 6 iOS 10.3.3 のメールアプリの本文で「こんに」と入力した時の予測変換

第4章 本研究の特徴

本研究では、入力データ保持を実現した操作支援システムの自動生成を、統合するアプリケーションの入力項目名を概念として含むオントロジと、アクティビティ図をもとに行う。なお、アクティビティ図には自動生成に必要な情報を追加記述する。本研究の特徴は以下の通りである。

4.1 入力再利用

ユーザが間違った手順に進んでしまい、正しい手順に戻るときに、間違った手順のときに入力してしまった内容を正しい手順で使えることが重要である。本研究が提案する操作支援システムの入力再利用機能では、ユーザが以前入力したデータをラベル名が類似する別の入力項目において再利用を行うことができる。また、どの程度まで再利用を行うかの基準の設定をユーザが行うことができる。具体的には、ユーザの設定に合わせて、ラベル名の類似度が低い場合でも再利用を可能に、あるいは、類似度が高くなければ再利用できないようにするといったことも可能である。なお、本研究では、ユーザが操作を間違え、そこから正しい手順に進んだ時に再利用を行うことを想定しているが、それに限らず、操作を間違えた場合でなくとも再利用は可能である。図 7 はアプリケーションの画面遷移図の例である。矢印の向きに画面遷移が行われ、画面 A がトップ画面（アプリケーションにおけるユーザが最初に操作する画面と定義する）である。画面 B ではユーザが入力を行い、画面 C での入力と選択によって画面 D、F にそれぞれ分岐する。ユーザの行いたい作業のゴールが画面 E、ユーザの現在画面が画面 G である場合を考える。図 7 の事例において、画面 F と画面 D で入力内容が共通である部分があるとする。このとき、画面 D ではまだユーザは操作を行っていないが、画面 F で行った入力を再利用することにより、画面 D での入力操作を削減することができる。

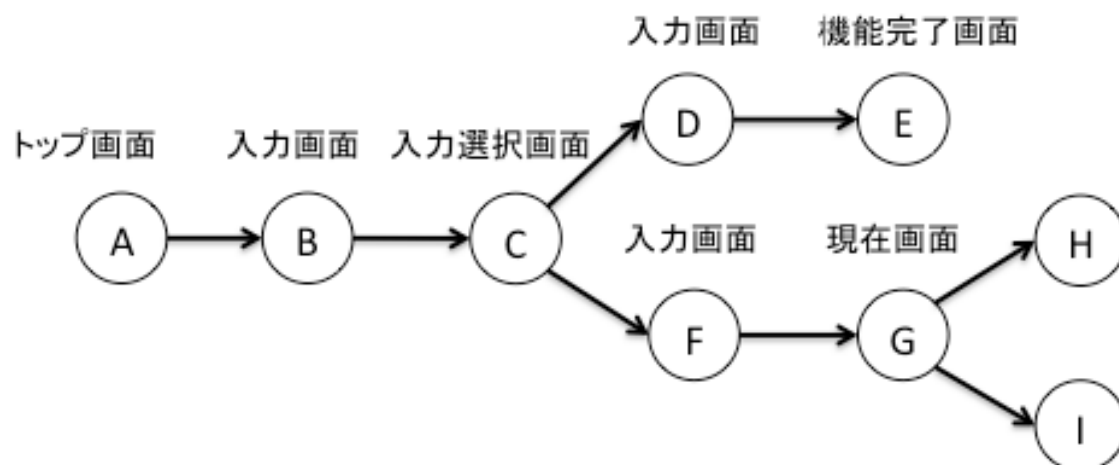


図 7 アプリケーションの画面遷移図の例

4.2 ユーザの状況に合わせた操作方法の提示

従来の操作支援は、機能の最初から最後まで操作方法を提示するものが多い。そうすると、一度行った入力を再度入力しなければならない。そのために、本手法では、現在の位置から、本来ユーザが利用したい機能の操作ステップまで戻ることを案内する。そうすることで、本来の目的ではない機能の操作手順に進んでしまったとき、目的の機能との共通手順での操作をしなくて済む。図 7 の事例を考えると、トップ画面 A から完了画面 E、トップ画面 A から現在画面 G までの経路（画面遷移の流れ）の共通部分は画面 A、B、C である。現在画面 G からトップ画面 A に戻ってしまうと、画面 B や画面 C における入力が破棄される。そして、完了画面 E まで辿っていくためには再度画面 B や画面 C にて同じ入力をしなければならない。現在画面 G から分岐点 C まで戻り、そこから完了画面 E までたどる手順を提示すれば、再度画面 B にて入力を行わずに済む。

4.3 開発者のコスト削減

本研究が提案する操作支援システムは、入力再利用機能と操作手順提示機能を持つ。入力再利用機能においては、オントロジの作成と、入力項目とラベル名の対応付けを開発者が行うことで自動生成できる。オントロジとは、概念や概念間の関係を表すモデルである。いくつかの研究では、要求分析の段階でドメインオントロジを用いることがしばしば提案されており[15]、要求分析の段階

でドメインオントロジを作成していれば、入力再利用機能の生成のコストを削減することができる。操作手順提示機能は、生成に必要な情報を追加記述したアプリケーションのアクティビティ図を基に自動生成される。UML を用いたアプリケーションを開発する際、開発者はアプリケーションに関するアクティビティ図などの UML モデルを作成することが多い[16]。よって、操作支援システムを生成する際の開発者の手間としては、自動生成に必要な情報を作成したアクティビティ図に追加記述するだけで良い。追加記述が付加されたアクティビティ図を基に、既存のアプリケーションに操作支援システムのコードを生成する。

4.4 学習効果

本研究が提案する操作支援システムは、ユーザが操作手順を間違えたときの学習効果が高く、ユーザが同じ操作をする次の機会にスムーズに操作できるようになることが期待できる。操作した手順を戻ることにより、どこで間違えたかをユーザは確認することができる。しかし、従来の操作支援システムでは、常に最初からの手順を提示するため、ユーザはどこで間違えたかを理解することなく、目的までの操作をしていくことになる。人間は失敗することで多くのことを学ぶことができる[17]。本システムは、アプリケーションの操作において、この失敗から操作手順を学ぶことを実現している。

第5章 操作支援システムの動作

5.1 操作支援システムの利用

本節では、図 7 のアプリケーションを適用事例にユーザから見た操作支援システムの動作を述べる。図 8 は、図 7 のアプリケーションの画面遷移をアクティビティ図で表したものである。なお、このアクティビティ図でのアクティビティは画面表示のみになっている。適用事例は 4.1、4.2 節で述べたものと同じ状況を考える。ユーザの現在画面は画面 G、ゴールは画面 E である。このアプリケーションでは、

- 機能 1：画面 A から画面 E までの手順
- 機能 2：画面 A から画面 H までの手順
- 機能 3：画面 A から画面 I までの手順

が使えるものとする。つまり、ユーザは間違って機能 2、あるいは 3 の途中にいるが、本来は機能 1 を利用したいということである。また、表 1 はこのアプリケーションで用意されている入力項目間での一致度を表したものである。なお、この一致度はオントロジの概念間の類似度計算によって求められたものである。一致度は 0 から 1 までの範囲で値を取り、数値が高ければ高いほど、より類似しているということになる。類似度の計算は後に 5.2.2 節にて記述する。画面 F には入力項目 1、画面 D には入力項目 2、画面 E には入力項目 3 があるとする。

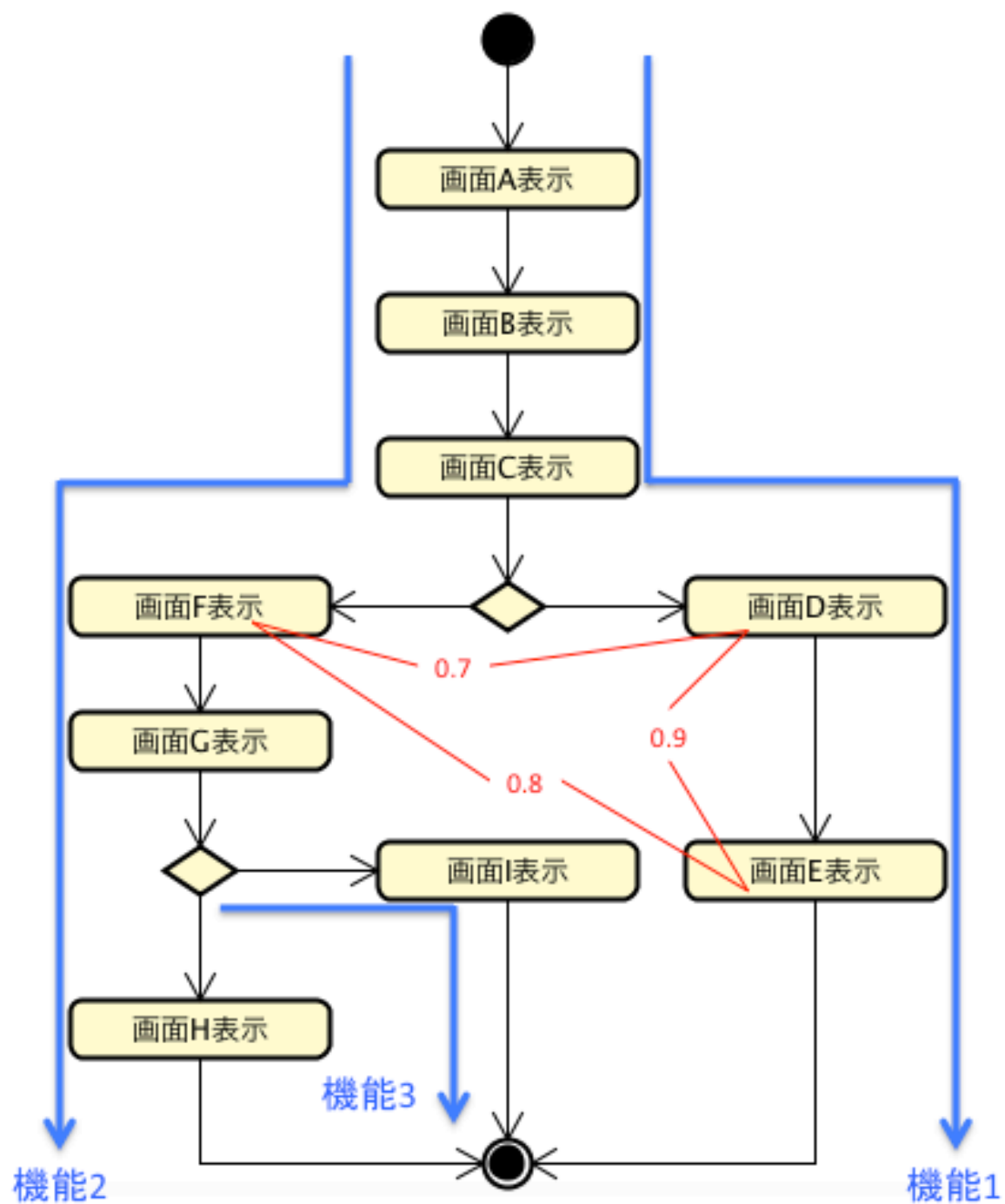


図 8 アプリケーションの画面遷移図

表 1 入力項目の一致度

	入力項目 1 (画面 F)	入力項目 2 (画面 D)	入力項目 3 (画面 E)
入力項目 1 (画面 F)		0.7	0.8
入力項目 2 (画面 D)	0.7		0.9
入力項目 3 (画面 E)	0.8	0.9	

5.1.1 ユーザによる入力再利用

5.1 節で述べた状況において、本研究で提案する操作支援システムをアプリケーションに適用し、ユーザが入力再利用機能を利用する場合を考える。アプリケーション起動と共に操作支援システムが立ち上がり、そこで再利用を行うかどうか判断するための閾値をユーザが決定する。閾値は 0~1 までの値を取り、ラベル名間での類似度を表す。各入力項目の意味はラベル名によりユーザに伝えられるため、入力項目はラベル名と対応付けがなされている。これにより、各ラベル名の類似度はそれに対応する入力項目の類似度となる。数値が大きいほど類似性の高い入力データの再利用を行うが、再利用が行われる範囲は狭くなる。逆に、数値が小さければ、類似性の低い入力データの再利用も行われるが、再利用が行われる範囲は広くなる。ユーザが閾値を決定した後、ユーザが今までに入力した内容と類似したラベル名を持つ入力項目を含む画面に遷移した際に、ユーザが決定した閾値に応じて入力の再利用が自動的に行われる。ユーザが機能 1 を利用するためには、画面 C まで戻り、そこから画面 D、画面 E と遷移し、それぞれの画面内にある入力項目 2、3 で新たにデータを入力する必要がある。ユーザはすでに画面 F にある入力項目 1 を入力しているため、入力項目 2、3 にて再利用を行うことが可能である。仮に、ユーザがアプリケーション起動時に操作支援システムにて設定した閾値を 0.6 に設定したときは表 2 のように、閾値を 0.8 に設定したときは表 3 のようになる。

表 2 入力再利用（閾値=0.6 の時）

	入力内容
入力項目 1	再利用機能利用前に入力済み
入力項目 2	入力項目 1 を再利用
入力項目 3	入力項目 2 を再利用

表 3 入力再利用（閾値=0.8 の時）

	入力内容
入力項目 1	再利用機能利用前に入力済み
入力項目 2	入力再利用なし
入力項目 3	入力項目 2 を再利用

5.1.2 ユーザによる操作手順の参照

ユーザは本来のゴールを達成するための操作手順を知りたいときに、アプリケーション起動時に共に起動する操作支援システムにアクセスする。操作支援システムにアクセスすると、このアプリケーションで利用可能な機能一覧がユーザに提示される。また、操作支援システムでは、必要に応じて、キーワードに関連したゴールに絞り込むことが可能である。ユーザが機能一覧から機能を選択すると、ユーザの現在画面からその機能のゴールまでの提示文（ユーザに提示される機能の利用方法に関する情報）がユーザに対して提示される。この提示文は箇条書きになっており、一から順番に操作を行なっていく。箇条書きの一つにつき、一画面での操作となっている。

5.2 入力再利用の仕組み

5.1.1 節では、ユーザから見た操作支援システムの入力再利用を述べた。本節では、操作支援システムによる入力再利用の内部処理について述べる。本研究で提案している入力再利用はユーザのこれまでの入力データを他の画面の入力項目において再利用を行うものである。よって、入力再利用を実現するのに必要とする処理は以下のものとなる。

- (1) 入力データの保存
- (2) 入力データ再利用の判断
- (3) 入力データの自動補完

5.2.1 入力データの保持

入力データを再利用するためには、これまでの入力データを保管しておかなければならない。入力データを保持する際には、本システムでは入力データとラベル名の対応付けを行う。対応付けを行う仕組みとして、入力項目とラベル名の対応付けを開発者が行う必要があるが、これについては後に 6.2 節で記述する。例として、“Last Name”という入力項目があり、ユーザが“Nakamura”と入力を行なってコミットしたとする。すると、入力再利用機能は“Last Name = Nakamura”というように、“Last Name”というラベル名と“Nakamura”という値をセットで保持を行う。保持するタイミングとしては、ユーザがコミットしたタイミングが最も入力データが正しい可能性があるため、ユーザが入力データをコミットした時に入力データの保持が行われる。

5.2.2 入力データの再利用

入力再利用機能はこれまでの入力データを他の画面内の再利用可能な入力項目において再利用を行うものである。画面遷移直後に再利用を行うが、その際に再利用可能かどうかを判断するために、本研究では各入力項目のラベルの基本概念を持ったオントロジを利用する。基本概念とは、他の概念に依存せずに自身の性質のみで決まる概念のことである。画面遷移をした際に、その画面内に含まれる各入力項目と、保管されている入力データのラベルがどれほど類似しているのかを計算し、ユーザが設定した閾値を超えていれば再利用できると判断を行う。この類似度は、オントロジ内でそれぞれの概念が持つ属性が一致しているかどうか、オントロジの階層の深さによって決まる。入力再利用可能かどうかの判断の後に、もし再利用を行うことに決定すれば、これまでの入力データを再利用可能な入力項目において自動補完を行う。

5.3 操作手順提示の仕組み

5.1.2 節ではユーザから見た操作支援システムの操作手順提示機能の動作を述べた。本節では、操作手順提示機能の内部処理について述べる。本研究で提案

している操作手順提示機能はユーザのアプリケーション内での現在画面から、ユーザの求めるゴールまでの手順を提示する。よって、操作手順提示機能が必要とする処理は以下のものとなる。

- (1) トップ画面から全ての機能完了までの経路導出
- (2) ユーザの現在画面の把握
- (3) ユーザの現在画面から機能完了までの手順の導出

5.3.1 トップ画面から機能完了までの経路導出

操作支援システムが起動した際に（アプリケーションを起動したとき）、アクティビティ図を解析して得られたデータをもとに、トップ画面から全ての機能の完了までの経路を最初に導出する。全ての経路を導出すると、操作支援システムは終了するまでその経路情報を保持し続ける。図 8 のアクティビティ図における、アプリケーションの経路は、3 通りである。この 3 通りの経路のアクティビティ図を図 9、10、11 に示す。図 9 は、機能 1 を利用するためのトップ画面から機能完了までの経路である。図 10 は、機能 2、図 11 は機能 3 を利用するためのトップ画面から機能完了までの経路である。

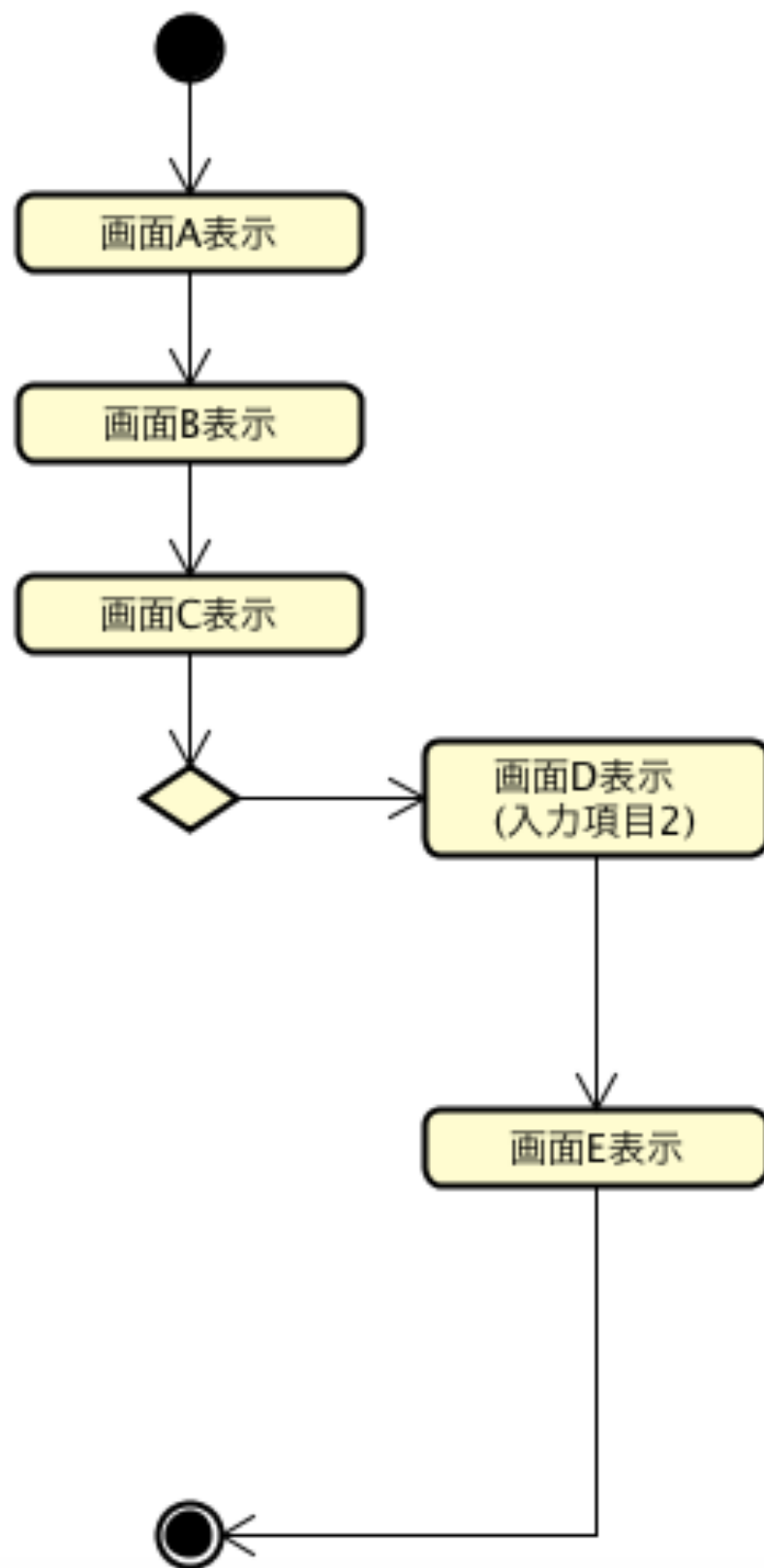


図 9 機能 1 の経路

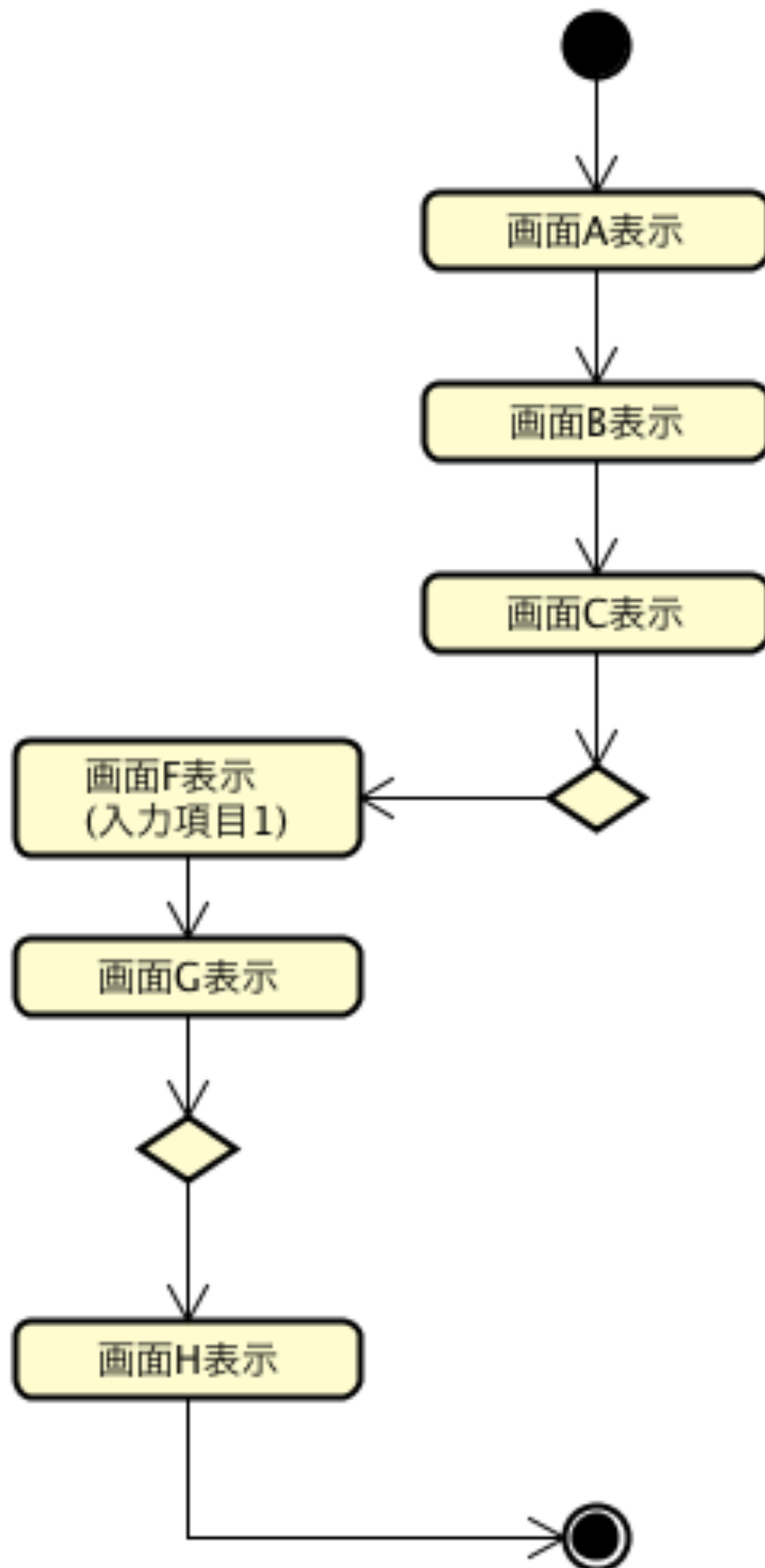


図 10 機能 2 の経路

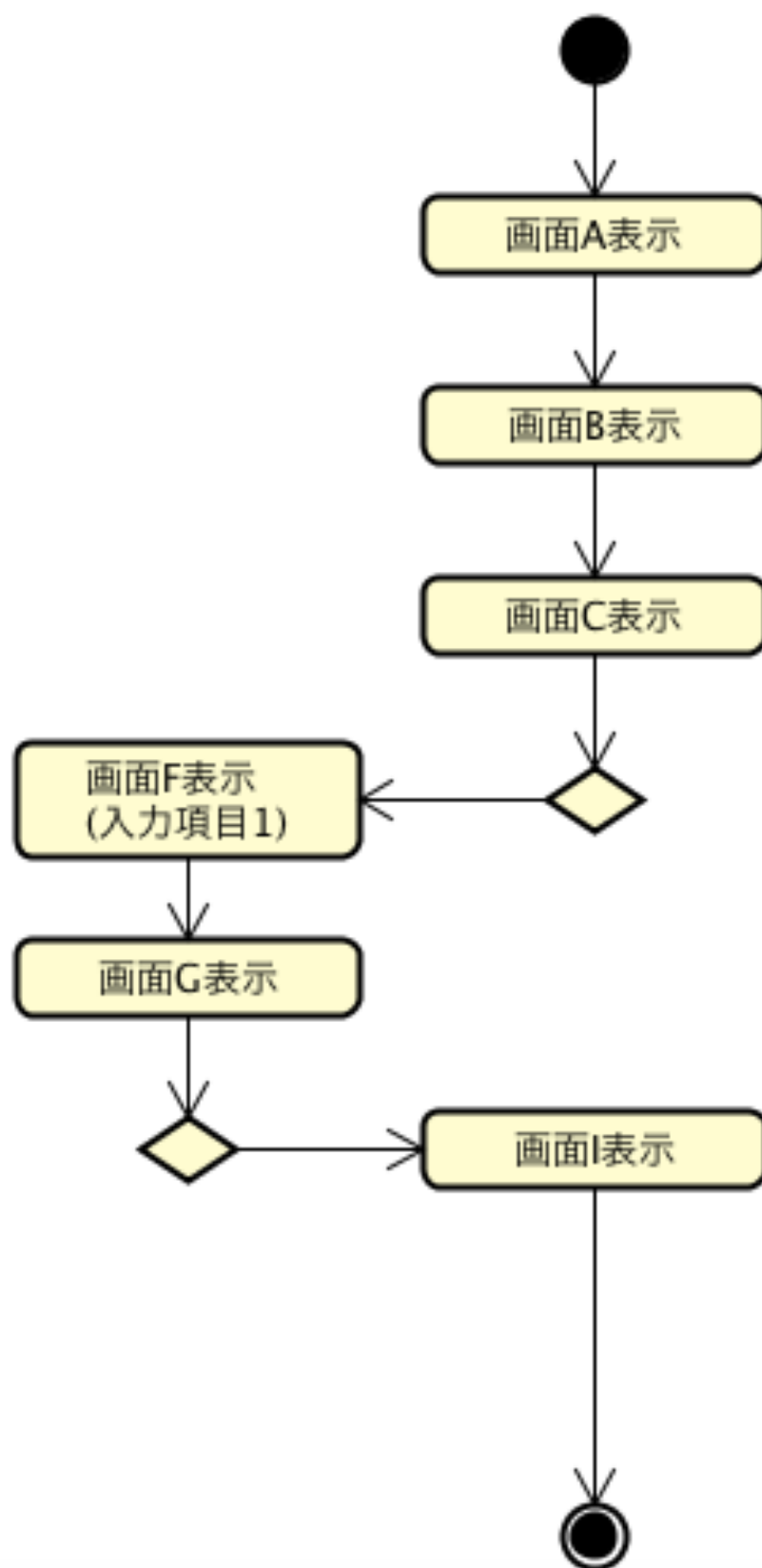


図 11 機能 3 の経路

5.3.2 ユーザの現在画面の把握と履歴保持

本研究で提案する操作手順提示機能は、ユーザの現在画面からゴールまでの手順を求めるものとなるので、ユーザの現在画面を把握する必要がある。アプリケーションを起動したとき、操作支援システムは最初にアプリケーションのトップ画面をユーザの現在画面として情報を保持する。そして、ユーザが画面遷移すると、操作支援システムは新たにアクティブになった画面を感知する。この感知した画面を記録していくことにより、画面遷移の履歴を生成していく。それぞれの画面には、識別子があらかじめ割り振られているので、その識別子をもとにどんな画面であるのかを認識することができる。また、ユーザの現在画面は最後にアクティブであることを感知した画面となる。このようにして、操作支援システムは常にユーザの現在画面の情報を保持する。例えば、図 8 のアクティビティ図でのトップ画面は画面 A であり、ユーザが画面 B に遷移すると、ユーザの現在画面は画面 B、履歴は A→B となる。

5.3.3 ユーザの現在画面からゴールまでの手順の導出

ユーザが操作支援システムにアクセスすると、システムはユーザの画面遷移の履歴をもとにトップ画面からユーザの現在画面までの経路を導出し、その情報を保持する。次に、操作支援システムはユーザの画面遷移の履歴と、トップ画面から選択された機能のゴールまでの経路の重なりを確認する。重なりを確認するとき、以下の 2 通りのパターンが考えられる。

(1) トップ画面から機能完了までの経路にユーザの現在画面が含まれる場合

トップ画面から機能完了までの経路の中にユーザの現在画面が存在する場合、トップ画面から現在画面までの経路は、トップ画面から機能完了までの経路の一部である。よって、ユーザに対して提示する経路は両者の経路の共通部分でない経路である。ユーザの利用したい機能が機能 2 である場合、トップ画面から機能完了までの経路は図 10 になる。また、ユーザの現在画面が画面 G である場合、トップ画面から現在画面までの経路は図 12 の点線矢印の経路のようになる。この時、現在画面からゴールまでの経路は共通部分でない経路、図 12 の太字矢印の経路のようになる。

(2) トップ画面から機能完了までの経路にユーザの現在画面が含まれない場合

トップ画面から機能完了までの経路の中にユーザの現在画面が存在しない場合、両者の経路の共通部分の最後の画面から分岐がなされる。よって、現在画

面から機能完了までの経路は、現在画面から分岐点まで戻った経路と、分岐点から機能完了までの経路を連結したものとなる。操作支援システムは、この連結した経路をユーザに対して提示する。ユーザの利用したい機能が機能 1 である場合、トップ画面から現在画面までの経路とトップ画面から機能完了までの経路の分岐点は画面 C となる。よって、現在画面から画面 C まで戻った経路と、画面 C からゴールまでの経路を連結したものをユーザに対して提示する。この連結経路は図 13 の太字矢印の経路のようになる。

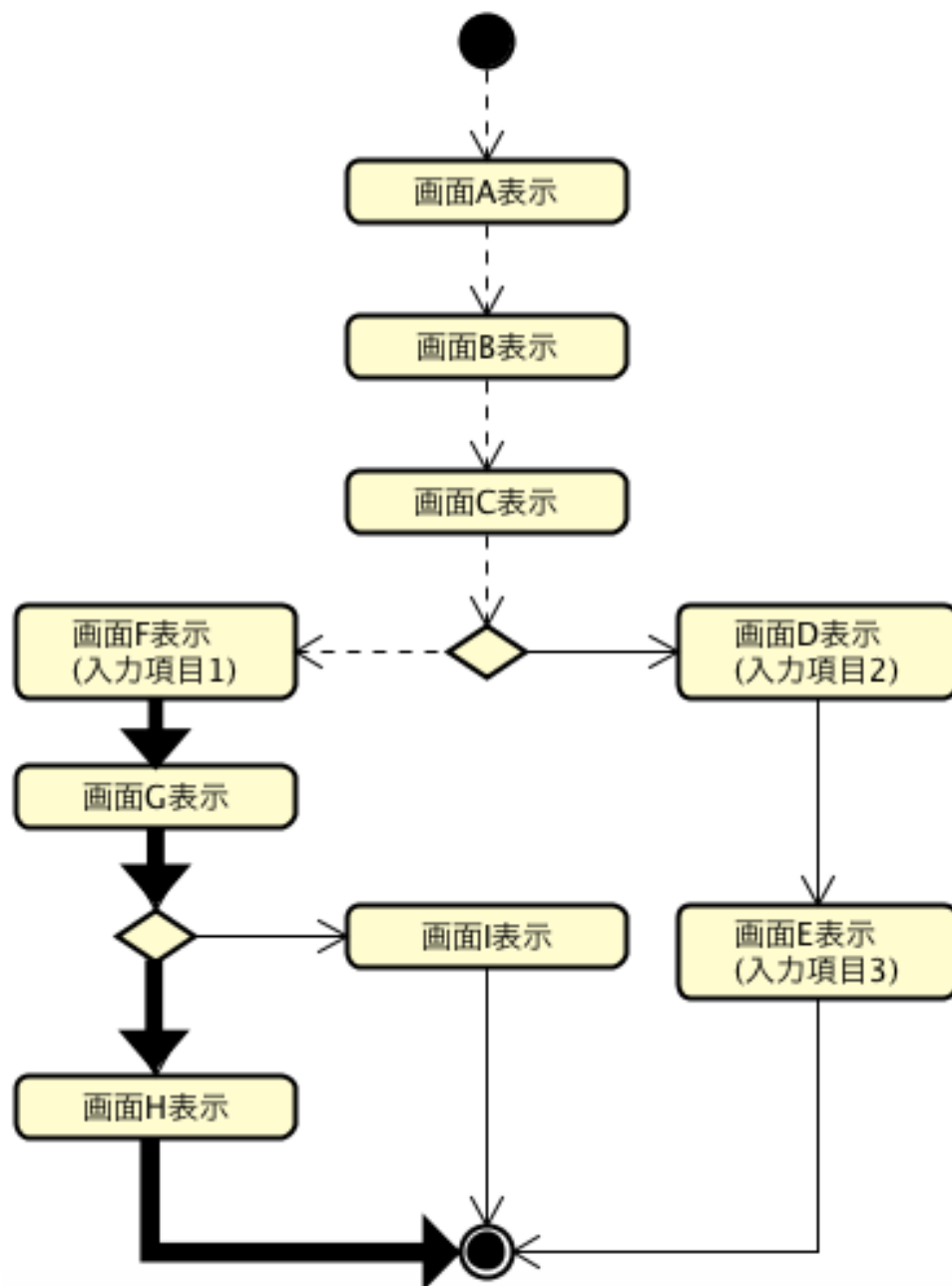


図 12 トップ画面から機能 2 完了までの経路と現在画面までの経路の共通部分(点線)と現在画面から機能 2 完了までの経路(太線)

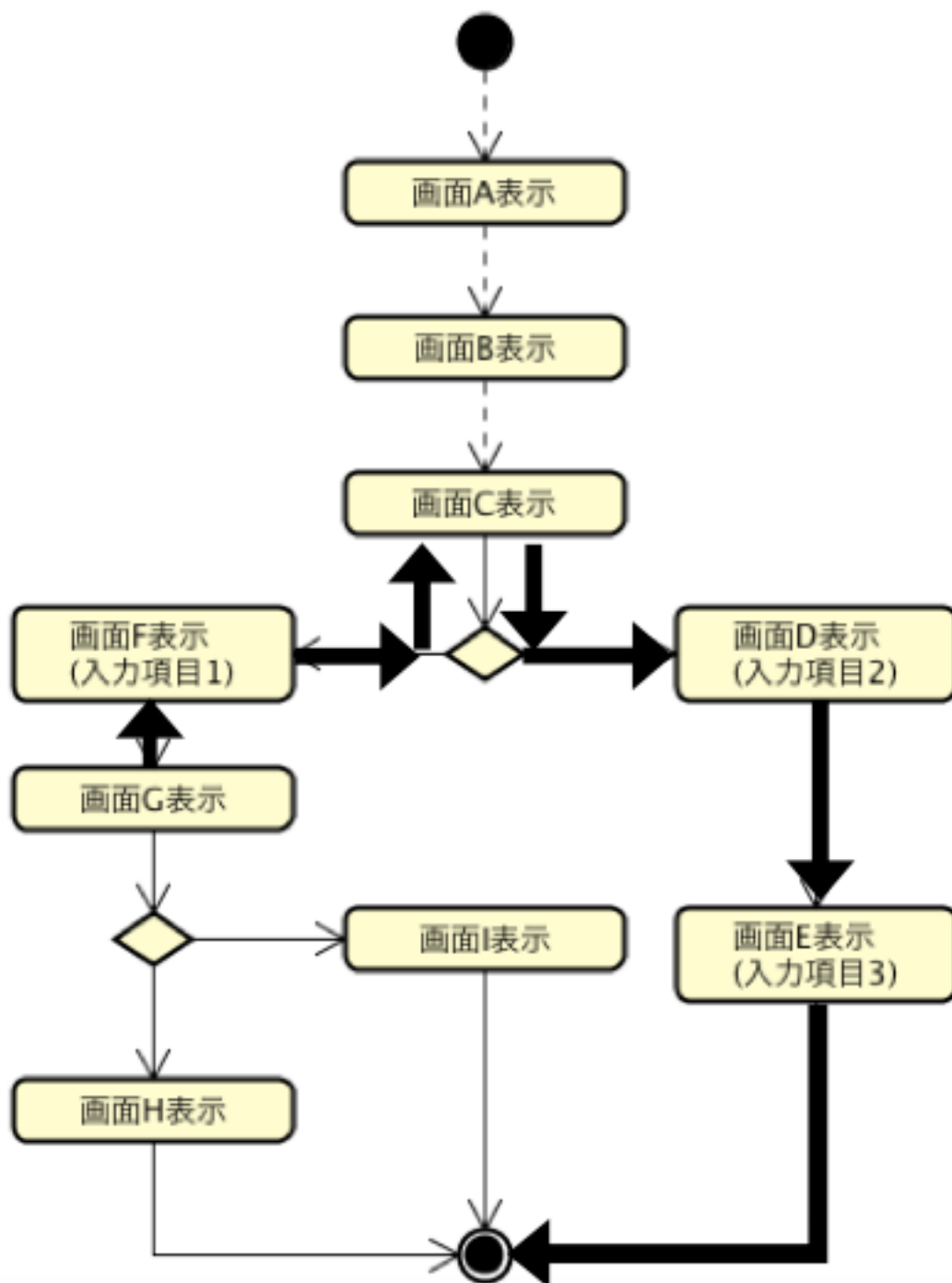


図 13 トップ画面から機能 1 完了までの経路と現在画面までの経路の共通部分(点線)と現在画面から機能 1 完了までの経路(太線)

第6章 本研究の詳細

操作支援システムの自動生成の手順は以下の通りである。

- (1) オントロジの作成
- (2) 入力項目とラベル名の対応付け
- (3) アクティビティ図の記述
- (4) アクティビティ図の解析
- (5) 開発者による機能名と提示文の記述
- (6) 操作支援システムのコード生成

なお、本システムにおいては、クライアント側が **HTML** と **JavaScript** で動き、サーバ側で **Java** を扱える **Web** アプリケーションを対象として、操作支援システムを実装している。本研究では、法造のオントロジエディタ[18]を利用することによって類似度の計算を行っている。ただし、オントロジのデータを扱うことができ、そのデータから入力項目間の類似度を計算すること、ラベル名とテキストフィールドとの対応付けができるのであれば対象アプリケーションのプログラミング言語や環境はこれに限らない。

6.1 オントロジの作成

前述したように、入力再利用機能での再利用判断のために、オントロジを用いる。オントロジの作成は開発者が行う必要がある。オントロジを作成する際には、入力項目のラベル名を持った概念を全て含ませる必要がある。また、各入力項目のデータタイプを設定することで、入力項目間の類似度をより正確に計算することが可能となる。もし入力項目間で関係付けが可能な場合は、関係づけを行う必要がある。

図 14 は法造オントロジエディタで作成したオントロジの事例である。”User Number”と”Manager Number”という入力項目があれば、下図のように基本概念として定義する。どちらも社員番号を表す入力項目であるならば、新たに

number 型の属性を持った”Personnel Number”という基本概念を定義し、”User Number”と”Manager Number”の親の概念とする。このように定義することで、類似度をより正確にすることができる。

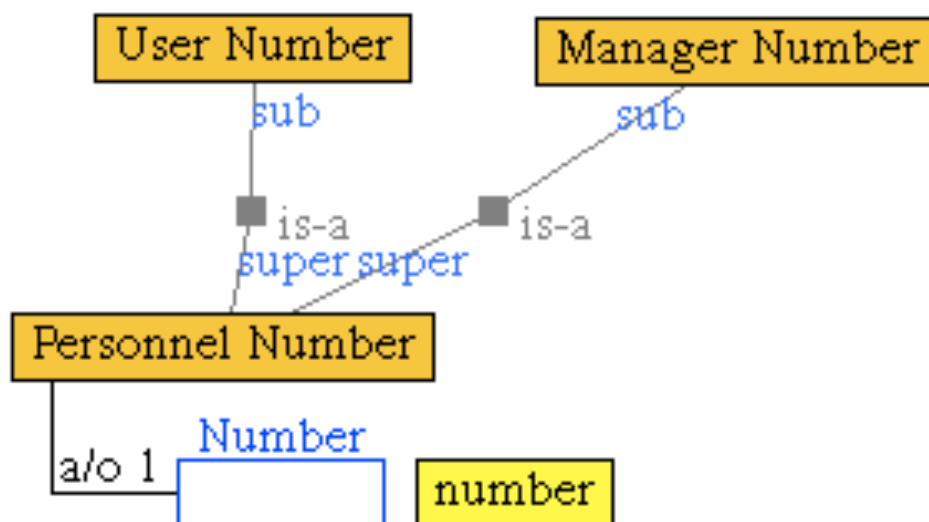


図 14 オントロジの作成例

6.2 入力項目とラベル名の対応付け

5.2.1 節にて述べたように、入力再利用機能が入力項目に入力された値とラベル名を対応づけて保持するためには、事前に開発者が対応付けを行っておく必要がある。対象アプリケーションに本システムを組み込むことを想定しているため、本システムの入力再利用機能がプログラマ的に読み解くことができるように対象アプリケーションのプログラム内で対応付けを行う必要がある。また、対応付けする際のラベル名は、オントロジで構築する際のワードと同じにする必要がある。本システムでは HTML の独自データ属性を利用することで、入力項目とラベル名を対応付けしている。図 15 は、実際に入力項目とラベル名を対応づけている例である。「E-mail」を表す入力項目の input タグ内に、独自データ属性「data-label」を置き、値として「E-mail」を与えている。

```

<tr>
  <td width="25%" class="ExpenseTagOffset">E-Mail:</td>
  <td width="75%" class="offsetColor"><input type="text" data-label="E-Mail"
    size="30"
    essType="String" essRequired="Yes" essTitle="E-mail" maxLength="60"></td>
</tr>

```

図 15 独自データ属性を用いた入力項目とラベル名の対応付け

6.3 アクティビティ図の記述

本システムでは対象とするアプリケーションのアクティビティ図を解析する。アクティビティ図は制御のフローを表すことができ、画面遷移を表すのに適している。また、アクティビティ図には操作支援システムの自動生成に必要な情報を追加記述する。

6.3.1 アクティビティ図構成要素

本システムの自動生成に必要なアクティビティ図の主な構成要素を以下に挙げる[19]。本研究では、アクティビティ図を画面遷移図として扱う。

- 開始ノード
アプリケーションの画面遷移の開始を表すノードである。アクティビティ図では、黒丸で開始ノードを表現する。
- 最終活動ノード
アプリケーションの画面遷移の終了を表すノードである。アクティビティ図では、白丸の中に同心円の小さな黒丸を書いたもので表す。
- アクティビティ
アプリケーションの画面表示処理を表す。アクティビティ図では、角が丸くなった長方形で表現され、その中に画面表示処理の記述をする。
- 制御フロー
アクティビティや様々なノードへのフローである。アクティビティ図では、

矢印で表現される。

- デシジョンノード

アプリケーションの画面遷移で分岐が起こる地点である。アクティビティ図では、菱形で表現される。例えば、図 8 のアプリケーションでは、画面 C での操作によって機能 1 と機能 2 に分岐する。このように、分岐によって複数の遷移が起こる場合はデシジョンノードが必要である。

- マージノード

アプリケーションの画面遷移で分岐したフローが合流する地点である。アクティビティ図では、菱形で表現される。一つの画面に遷移するフローが複数ある場合は、マージノードが用いられる。

6.3.2 アクティビティ図への追加記述

操作手順提示機能では、対象とするアプリケーションの画面遷移の情報を得るために、アクティビティ図を解析する。また、画面遷移の情報を得るために、アクティビティ図には、アクティビティ図の基本的な構成要素に加え、操作支援システムの自動生成に必要な情報である画面オブジェクト名を追加記述する。画面オブジェクト名とは、アプリケーションのソースコード内で宣言されている画面の役割を担うオブジェクトの名前である。この追加記述によって、追加記述がされたアクティビティは画面表示のアクティビティであることを表すことができ、アクティビティ図を画面遷移図として扱うことができる。

アクティビティ図内のアクティビティに画面オブジェクト名が対応付けられることにより、対応付けられたアクティビティが画面表示のアクティビティであることを示す。また、この対応付けられた画面オブジェクト名が各アクティビティの識別子の役割を果たす。対応付ける方法としては、対応付けられるアクティビティの元の記述に対して「*Top_Window*」のように、「*」で囲って追加記述をする。追加記述したアクティビティ図は図 16 のようになる。

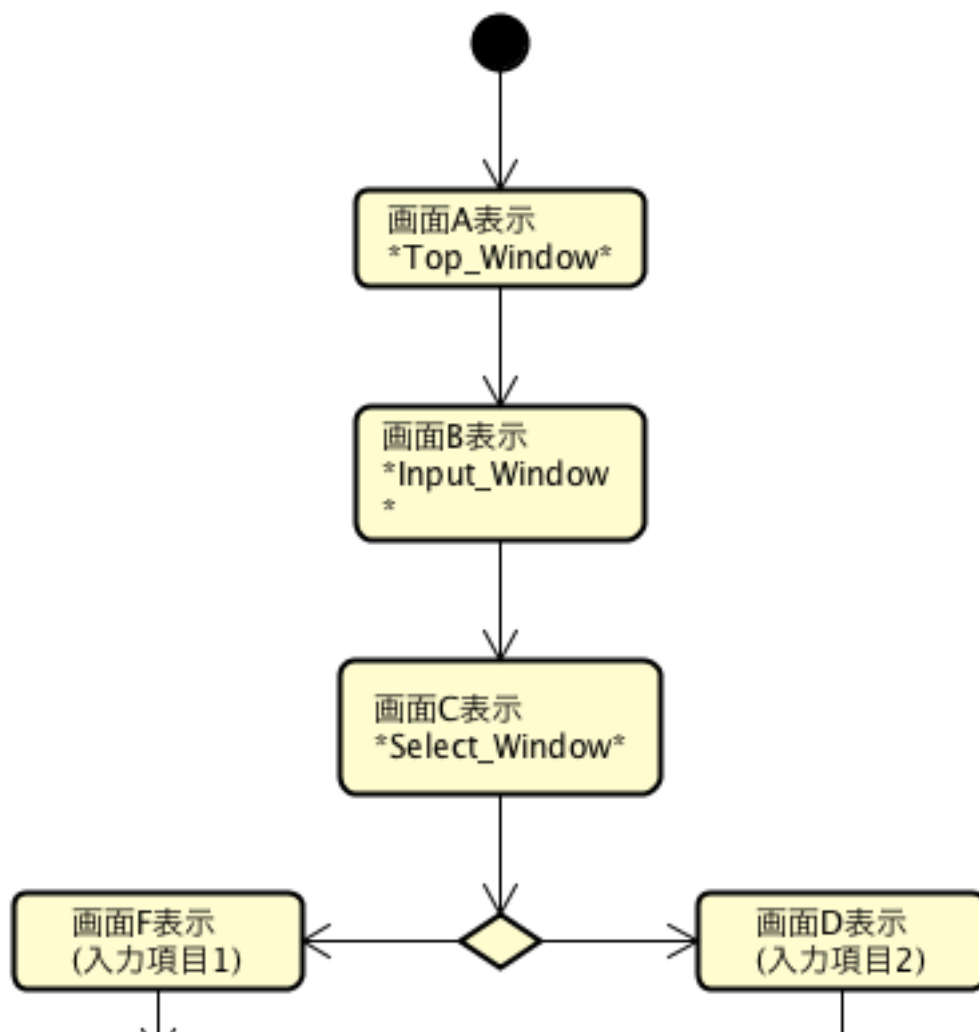


図 16 追加記述を行ったアクティビティ図

6.4 アクティビティ図の解析

図 16 のように追加記述が行われたアクティビティ図をもとに操作支援システムのコードの自動生成に必要な情報を抽出する。アクティビティ図から抽出可能な情報として、以下のものが挙げられる。

- 画面オブジェクト名

アプリケーションソースコード内で宣言されている各画面のオブジェクト名である。アクティビティ図には、それぞれのアクティビティに該当する画面オブジェクト名が紐づけられており、これらの画面オブジェクト名を抽出する。また、各画面オブジェクトには画面 ID を生成するので、画面オブジェクト名を

抽出する必要がある。

- 画面遷移情報

基本的には、画面遷移は木構造になっている。この木構造を抽出するために画面オブジェクト名を追加記述したアクティビティ図が用いられている。画面オブジェクト名が紐づけられたアクティビティのみに絞り、画面遷移の情報を抽出する。

また、自動生成システムは解析された情報をもとに、考えうるアプリケーションの全ての経路を導出し、これら全ての経路情報を開発者に対して提示する。

6.5 開発者による機能名と提示文の記述

操作支援システムで利用される機能名や提示文の内容を決定する必要がある。そのために、アクティビティ図の解析により得られた全ての経路情報を開発者に提示し、これらの経路情報に対して開発者は以下の記述を行う。

- 機能名

6.4 節の解析によって得られた全ての経路情報に対して、それぞれの経路がアプリケーション内で何を行うためのものであるかについての記述を開発者が行う。この記述された内容は、ユーザが本操作支援システムを利用したときに表示される機能一覧として利用される。

- 提示文

6.4 節の解析によって得られた経路情報に対して機能名を記述した後、この経路を辿るために各画面においてユーザが行うべき操作を、各画面に対して開発者が記述を行う。ここで記述された内容は、ユーザが本操作支援システムを利用したときに表示される提示文として利用される。本研究では、ユーザが何を入力し、選択肢では何を選択するのかなど、機能を利用するための条件について開発者が詳細に記述することを想定し、ユーザがそれぞれの機能を使うための条件を学習しやすくする。

これらの情報が追加された経路情報を操作支援システムは保持する。

6.6 操作支援システムのソースコード生成

操作支援システムを実現するためのソースコードは、6.4、6.5 節で述べた情報をもとに生成を行う。ソースコードの生成では、操作支援システムのインタフェース、入力再利用を行う機能、現在画面の把握と履歴保持を行う機能、経路導出を行う機能が含まれる。

- (1) 操作支援システムのインタフェース
- (2) 入力再利用を行う機能
- (3) 現在画面を把握する画面識別コード
- (4) 経路導出

6.6.1 操作支援システムのインタフェース

入力再利用を行うためには、入力再利用の判断に用いる閾値をユーザに問い合わせる必要がある。よって、問い合わせのためのインタフェースを実現するソースコードを生成する。ユーザはいつ如何なる時でも操作支援システムにアクセスできるようにする必要がある。従って、本システムはヘルプボタンを画面遷移に関係なく常にアプリケーション上に配置するコードを生成する。あるいは、Web アプリケーションであれば、新たに本システムにアクセスするためのタブを生成する。また、ユーザが操作支援を受けるためのインタフェースが必要であるので、操作支援システムのインタフェースを表す新たな画面オブジェクトを生成する必要がある。これらは、オントロジやアクティビティ図から得られる情報に関係なく生成するものである。

6.6.2 入力再利用を行う機能

5.2 節で述べたように、オントロジを読み込むことで入力項目間の類似度を計算し、入力再利用を求められたときに、ユーザからの閾値を受け取り、閾値を超える一致度の入力項目を再利用するソースコードを生成する。なお、本研究では、法造のオントロジエディタを利用することによって類似度の計算を行う。類似度の計算は、基本概念間でのみ行われ、概念間の関係と、各概念が持つ属性を照らし合わせて計算が行われる。

6.6.3 現在画面把握機能

本システムにおいて、ユーザの現在画面の把握と履歴保持を行う機能は、ユーザが画面遷移をする度にユーザの現在画面と画面遷移の履歴を更新する。6.4 節にて画面オブジェクト名を抽出することにより、画面オブジェクト名をそれぞれアプリケーション内の画面の識別子として扱う。画面の識別子の役割を担う画面オブジェクト名と 5.3.2 節にて記述したアルゴリズムをもとに現在画面の把握と履歴保持を行う機能のソースコードを生成する。

6.6.4 経路導出機能

経路導出を行う機能は、ユーザの現在画面からゴールまでの経路を導出し、ユーザに対して導出した経路を提示する。ユーザが操作支援システムにアクセスし、機能一覧から機能を選択した時に動作する。6.4 節にて解析され、6.5 節にて追加記述が行われた全ての経路情報と、5.3.3 節にて記述したアルゴリズムをもとに経路導出を行う機能のソースコードの生成が行われる。

第7章 評価

本研究の有効性を評価するために、被験者による実験を行った。評価のために、本研究で提案する操作支援システムの対象アプリケーションとして、入力項目が多く、再利用可能な入力項目があるオープンソースの経費レポート提出システム[20]を選んだ。

7.1 被験者

8人の学生に被験者として協力してもらった（全員男性）。被験者は全員情報工学を専攻しており、被験者の平均年齢は22.6（21～24）であった。全ての被験者は、パソコンの操作には慣れているが、今回の経費レポート提出システムに触った経験はない。被験者は2つのグループに分けられ、グループA（4人）が操作支援システムを使いながらのタスクを行なってもらった後に、経費レポート提出システム付属の操作マニュアルを使いながら、1回目と同じタスクを行なってもらった。グループB（他の4人）では、グループAのタスクの流れと逆である。

7.2 手順と装置

グループA、B共に、操作支援システムと経費レポート提出システムの概要を説明した後にタスクを行なってもらった。操作支援システムを用いてタスクを行う際には、入力再利用で利用する閾値（0～1）の概要についても説明し、被験者に閾値を入力してもらってからタスクを行なった。被験者は経費レポート提出システムに馴染みがないため、例えば、被験者のアルバイト先での管理者業務など、被験者にコンテキストを想定してもらい、かつ業務知識をあまり必要としないタスクを行なってもらった。タスクの流れとしては、経費レポート提出システムの管理者側としてログインし、部署の追加、システムを利用するユーザの追加、そして最後に、メールアドレスの追加（領収書の画像をEメールで送付できるようにするための機能）を行なった。2回目のタスクを完了した後に、アンケートに答えて実験を完了とした。アンケートの質問項目について

は以下に示す。このアンケートの Q1、Q5～Q7 では、5 段階のリッカート尺度を採択しており、1 はそう思わない、2 はややそう思わない、3 はどちらでもない、4 はややそう思う、5 はそう思う、を表す。Q2 でも 5 段階のリッカート尺度を採択しているが、1 は従来の操作支援寄り、2 はやや従来の操作支援寄り、3 はどちらでもない、4 はやや本システムの操作手順提示による操作支援寄り、5 は本システムの操作手順提示による操作支援寄り、を表す。Q3、Q4、Q8～Q11 に関しては、被験者による自由記述で回答してもらった。

Q1. 操作手順の提示は適切になされましたか？

Q2. 従来の操作支援と、現在画面からの手順を提示する操作支援はどちらが有用ですか？

Q3. Q2 でそう答えた理由を教えてください

Q4. 操作手順提示の支援はどのような場面で役に立つと思いますか？

Q5. 必要な入力欄に再利用はされましたか？

Q6. 適切なデータが再利用されましたか？

Q7. 設定した閾値は適切でしたか？

Q8. 適切だと感じた再利用項目は何ですか？

Q9. 再利用して欲しかったのに再利用されなかった入力項目は何ですか？

Q10. 不適切だと感じた再利用項目は何ですか？

Q11. 自由な感想を教えてください

なお、操作支援システムの操作手順提示機能で提示する提示文は、対象アプリケーションで提供されている操作マニュアルと同等の内容とし、かつ言語による有意差が出ないように英語で記述した。

本評価実験では、Apple の Macbook Air (Mac OS 10.12.3) を用いて行い、ブラウザは Firefox 54.0.1 であった。対象アプリケーションは JAVA、HTML、JSP、JAVASCRIPT で実装されており、926 個のファイルで構成され、ソースコードの行数は 166803 行であった。

7.3 結果

実験を完了した後、各被験者にアンケートを実施した。図 17 は入力再利用に関する質問項目 Q5～Q7 のアンケート結果である。X 軸はユーザが選択した閾値であり、Q5～Q7 の 3 個の質問に対して、それぞれの閾値での回答結果とな

る。図 17 を参照すると、閾値が 0.2 の時が全ての質問において最も低い結果となり、Q6 においては、閾値が 0.3、0.4 の時に最も高い結果が得られた。Q7 においては、閾値が 0.4、0.7 の時に最も高い結果が得られた。

表 4 は、被験者が設定した閾値と入力再利用との関係を示したものである。入力項目数に関しては、今回設定したタスクの入力項目には必須ではない入力項目も含んでいたため、それによって被験者間で差が出ている。表 4 によると、閾値が高くなるほど再利用項目数が少なくなり、不適切な入力再利用項目数が減っているのが分かる。代わりに、適切な再利用項目数も減り、再利用したかった項目数は増える傾向にある。

図 18 は操作手順の提示による操作支援のアンケート結果である。従来の操作支援と、本システムの操作手順の提示の操作支援との比較では、有用だと感じた理由としては、

“ユーザの現在画面に合わせて操作手順の提示内容が変化するため、分かりやすかった”

“余分な情報がなく、自分のすべき操作のみが提示されるため、分かりやすかった”

といったことがコメントとして挙げられる。

なお、どちらでもないと答えた被験者では、

“そもそも操作支援システムが提供する操作手順を参照しなかった”

といったコメントが挙げられており、普段から操作マニュアルなどによる操作支援を受けない被験者であった。

操作手順の提示による操作支援の有用な場面として、

“対象アプリケーションの操作手順が複雑な場合やユーザが初心者である場合には有用だと感じられる”

といったコメントがいくつか挙げられた。

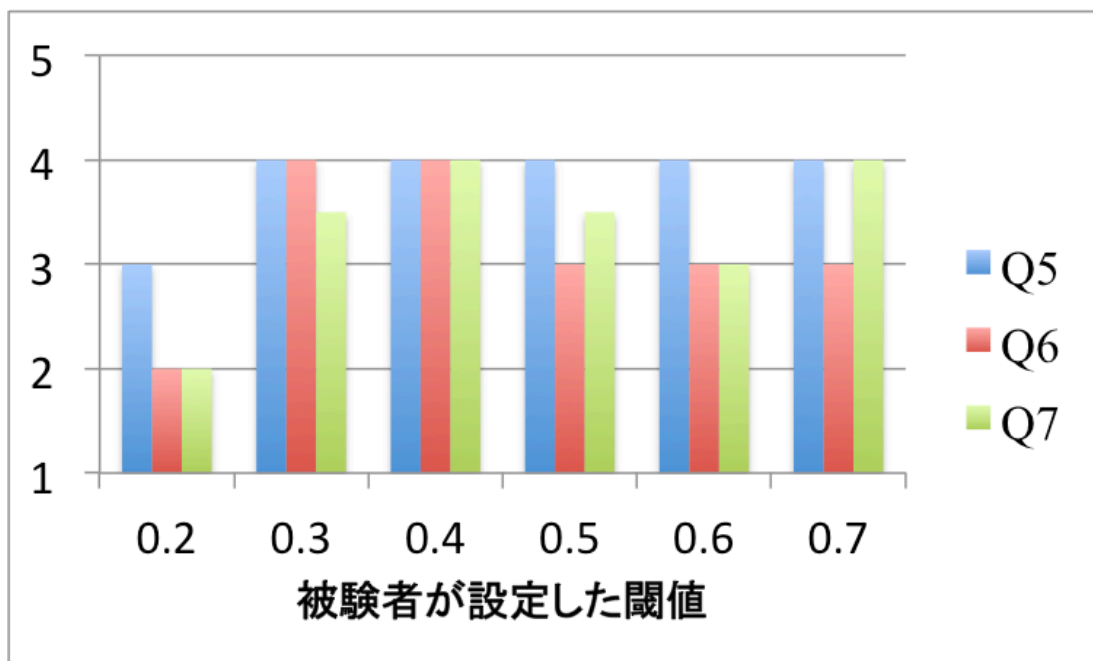


図 17 入力再利用機能に関するアンケート結果

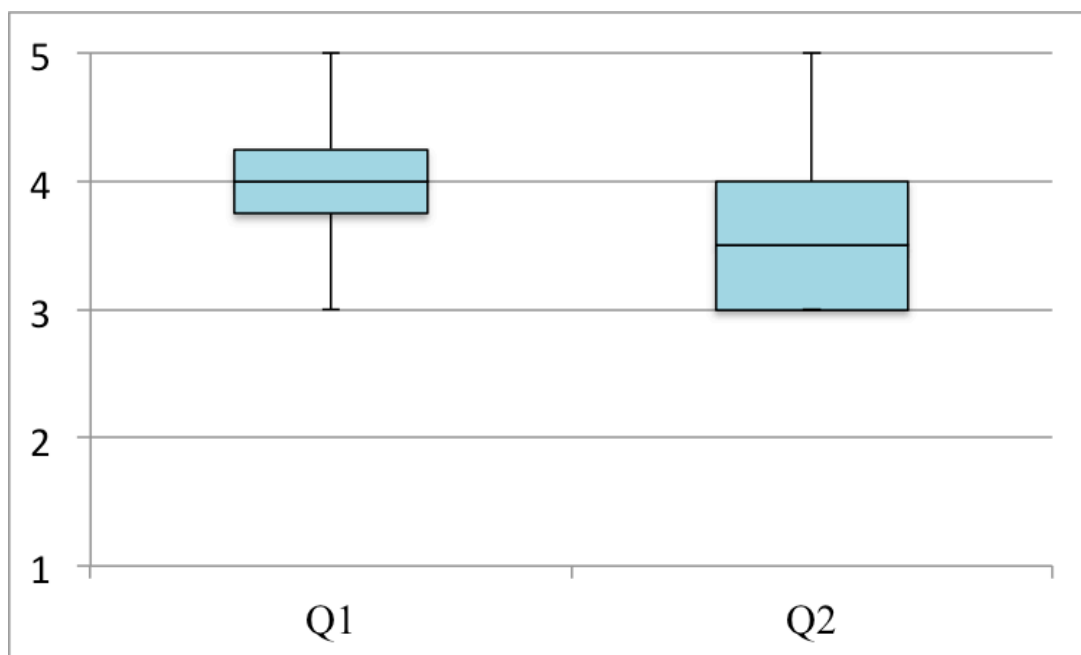


図 18 操作手順提示機能に関するアンケート結果

表 4 閾値と再利用の関係

閾値	入力項目	再利用項目	不適切な再利用 がされた項目	適切な再利用 がされた項目	再利用適用が 可能な項目
0.2	17	9	6	3	1
0.3	21	11	5	6	0
0.3	20	11	6	5	0
0.4	13	7	4	3	0
0.5	21	4	2	2	0
0.5	21	4	2	2	3
0.6	21	3	2	1	4
0.7	17	3	2	1	4

第8章 考察

7章の評価結果を踏まえて、この章では入力再利用と操作手順の提示による操作支援についてそれぞれ考察を行う。

まず、入力再利用による操作支援に関して、図 17 の結果を参照すると、閾値が 0.6 の時に、Q6、Q7 の結果が低くなっている。被験者の主観に影響するものではあるが、不適切な再利用が起きてしまえば、人によってはストレスに感じてしまうため、不適切な再利用を起こさないために閾値の基準を検討する必要がある。

表 4 を参照すると、同じ閾値で同じ再利用回数であるにもかかわらず、不適切な再利用項目数と適切な再利用項目数が異なる。これはユーザのコンテキストによって再利用が適切である場合も不適切である場合もあるからだと考えられる。今回の評価では、被験者がそのまま再利用されたデータを使用したり、あるいはそのデータを消して、別のデータを入力することがあったが、それはユーザがどのようなコンテキストを持ってタスクをこなすかにも依存すると考えられる。今回の評価では、ユーザのコンテキストを配慮せずに強制的に再利用が行われたが、ユーザのコンテキストを考えた上で再利用をするかどうかを決める必要がある。今後は、操作支援システム側がユーザのコンテキストを自動的に感知する構想、あるいは、ユーザ自身が操作支援システムにコンテキストを感知させるためのアクションをできるようにするための構想にすることで、解決ができると思われる。

また、閾値が上がるにつれ、再利用したかった項目数も共に増えている。これは、今回の評価で作成したオントロジの概念間の類似度が全体的に低く、その類似度よりも設定した閾値が高かったため、再利用がなされなかったからだと考えられる。今回の評価では、データタイプだけ設定した概念もあり、データタイプが同じで、かつ閾値が低ければ類似しない入力項目間で再利用が起こることがあった。例えば、ユーザの名前や部署の名前であればデータタイプは String 型となるが、それぞれの概念がデータタイプ以外定義されておらず、かつ閾値が低い場合は、ユーザの名前の欄に部署の名前が再利用されることがあり得る。より精密に類似度を測るために、各入力項目に関する概念をオントロジ内でより厳密に定義する必要がある。

操作手順の提示による操作支援に関して、図 18 を参照すると、被験者のネガ

ティブな結果はなかったが、操作手順の提示が非常に適切であった、あるいは従来のものよりも非常に有用であるといった結果は得られなかった。これは、今回の評価の対象アプリケーションの画面遷移の階層が広く浅かったため、操作手順の提示による支援があまり有用に感じられなかったためだと考えられる。コメントでもあったように、複雑な手順を持つシステムであればより有用だと感じられるものだと考えられる。

以上を踏まえ、いくつかの改善点は見受けられたが、被験者にとって適切だと感じられた入力再利用があったこと、浅い階層のアプリケーションに対しても現在画面からの操作手順の提示が効果的であったことから、本システムはユーザに対して効果的な操作支援を提供できたと言える。

第9章 おわりに

本研究では入力値再利用型操作手順提示システムの自動生成について提案している。本研究が提案する操作支援システムは、再入力や再操作をできる限り削減するために、これまでのユーザの入力データを類似した入力項目間で再利用を行う入力再利用機能と、ユーザの現在画面から機能完了までの手順を提示する操作手順提示機能を持つ。

本システムの生成には、対象アプリケーションが持つ入力項目を概念としたオントロジと、追加情報を記述したアクティビティ図をもとに行われる。追加情報として画面 ID や画面オブジェクト、ヘルプ情報が含まれる。

また、本研究では本システムの評価を行なった。結果として、入力再利用機能に関しては、閾値が低ければ再利用は適切にならない可能性が高いが、高くなっても適切にならない可能性もある、ということが示せた。これは、オントロジがどのように構築されているか、また、ユーザがどのようなコンテキストを持って操作しているかに依存したからだと考えられる。よって、より適切な再利用を行うために、オントロジの構成をより厳密に定義すること、操作支援システムにユーザのコンテキストを把握させることが必要だと思われる。

操作手順提示機能に関しては、従来のマニュアルと比べて非常に有用であることは示せなかったが、それは本研究で行なった評価実験で用いた対象アプリケーションの階層が広く浅かったためであり、より階層が深いアプリケーションで評価実験を行うことにより、本システムの効果性を示すことが期待できると思われる。

以上のことから、本研究の今後の課題は以下の通りである。

- 操作支援システムの改良
- 操作支援システムの再評価

謝辞

本研究を勧めるに当たり、御助言、御助力を頂いた早稲田大学理工学術院の深澤良彰教授に深く感謝致します。また、2年間に渡るご指導をして頂いた東京女子大学現代教養学部人間科学科の白銀純子准教授、神奈川工科大学情報学部情報ネットワーク・コミュニケーション学科の岩田一准教授に深く感謝致します。

参考文献

- [1] Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs, *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, Pearson, London, 2013.
- [2] 岩田一, 白銀純子, 深澤良彰, “開発情報からのチュートリアルシステム自動生成手法とその評価,” コンピュータソフトウェア, vol.24, no.3, pp.63-74, 2007.
- [3] Iwata Hajime, Fukazawa Yoshiaki, Shirogane Junko, “Generation of an operation learning support system by log analysis,” *Proceedings of 2nd International Conference on Software Engineering and Data Mining (SEDM)*, 2010.
- [4] Larry L. Constantine, Lucy A. D. Lockwood, “*Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design*”, Addison-Wesley Professional, 1999.
- [5] Han X. Lin, Yee-Yin Choong, Gavriel Salvendy, “A proposed index of usability: a method for comparing the relative usability of different software systems”, *Behaviour & Information Technology*, vol. 16, issue. 4-5, pp. 267-277, 1997.
- [6] Ahmed Seffa, Mohammad Donyaee, Rex B. Kline, Harkirat K. Padda, “Usability measurement and metrics: A consolidated model”, *Software Quality Journal*, vol. 14, issue. 2, pp. 159-178, 2006.
- [7] Parmit K. Chilana, Andrew J. Ko, Jacob O. Wobbrock, “LemonAid: Selection-Based Crowdsourced Contextual Help for Web Applications,” *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, 2012.
- [8] Parmit K. Chilana, Andrew J. Ko, Jacob O. Wobbrock, Tovi Grossman, “A multi-site field study of crowdsourced contextual help: usage and

- perspectives of end users and software teams,” Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI), 2013.
- [9] Tao Zhang, Ilana Stonebraker, “A Crowdsourced Library Help System,” Final Project Report for IMLS Grant Number LG-46-13-0239-13, 2014.
- [10] Sylvain Malacria, Joey Scarr, Andy Cockburn, Carl Gutwin, Tovi Grossman, “Skillometers: Reflective Widgets that Motivate and Help Users to Improve Performance,” Proceedings of the 26th annual ACM symposium on User interface software and technology., 2013.
- [11] 大山隆一郎, 山本喜一, “テキストエディタにおける能動的ヘルプシステム,” コンピュータソフトウェア / 日本ソフトウェア科学会, vol. 13, no. 5, pp. 371-386, Sept. 1996.
- [12] Joost Breuker, Radboud Winkels, Jacobijn Sandberg. “Coaching strategies for help systems: EUROHELP.” Proceedings of the 4th Annual ESPRIT Conference, 1987.
- [13] John W. Miler, Jossef Goldberg, “Text completion system for a miniature computer”, Microsoft Corporation. pp. US5896321, A. 1999.
- [14] Keith Vertanen, Haythem Memmi, Justin Emge, Shyam Reyal, Per Ola Kristensson, “VelociTap: Investigating Fast Mobile Text Entry using Sentence-Based Decoding of Touchscreen Keyboard Input”, Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI), 2015.
- [15] 長谷川亮, 佐伯元司, “ゴール指向要求分析のためのドメインオントロジーの利用法”, 電子情報通信学会知能ソフトウェア工学研究会(KBSE), vol. 107, no. 540, pp. 7-12, 2008.
- [16] MISCO オブジェクト指向研究会, オブジェクトモデリング表記法ガイド, 今野睦 (編) , プレンティスホール出版, 東京, 1998.
- [17] Sim B. Sitkin, “LEARNING THROUGH FAILUER: THE STRATEGY OF SMALL LOSSES,” Research in organizational behavior, vol. 14, 1992.

- [18] 児玉公信, UML モデリング入門, 日経 BP 社, 東京, 2008.
- [19] 大阪大学・溝口研究室:”法造オントロジー構築・利用の研究サイト”,
<http://www.hozo.jp/hozo/>, (accessed on 2017/9/12).
- [20] Expense Submittal System,
<https://sourceforge.net/projects/expense-ss/> (accessed on 2017/9/12).

研究業績

- 中村史門，岩田一，白銀純子，深澤良彰，“ユーザの状況を考慮した操作支援システムの自動生成”，電子情報通信学会知能ソフトウェア工学研究会（KBSE），2016年11月研究集会，2016.
- Shimon Nakamura, Hajime Iwata, Junko Shirogane, Yoshiaki Fukazawa, “Automatic Generation of an Operation Procedure Presentation System Reusing User’s Input Data”, Proceedings of the 2nd (HUCAPP 2018), Madeira, Portugal, January 2018.